

В документе 12 вопросов, которые касаются общей работы GDScript и игрового движка Godot и небольшое тестовое задание.

Ответьте пожалуйста развернуто на вопросы и, где это требуется, приведите примеры вашего кода, который поможет практически ответить на этот вопрос.

Вопросы

Вопрос 1.

Как работает система сцены в Godot и в чем ее преимущества?

Приведите блок схему простенького платформера на Godot, где есть:

- Стартовый экран загрузки с текстом
- Экран меню
- Игровой экран с персонажем и противников

Вопрос 2.

Объясните, как работают сигналы (Signals) в Godot?

Приведите общий пример использования сигналов и/или двух несвязанных между собой node (Не имеющие общего родителя)

Вопрос 3.

Как в GDScript организовать наследование, и зачем это нужно?

Вопрос 4.

Как работает система импорта ресурсов в Godot? Что произойдет, если изменить исходный файл изображения?

Вопрос 5.

Что такое `_process()` и `_physics_process()` в GDScript и чем они отличаются?

Вопрос 6.

Как создать и использовать таймер (Timer) в Godot?

Вопрос 7.

Объясните, как работает система слоев и масок (Layers and Masks) для коллизий в Godot.

Вопрос 8.

Как в GDScript организовать взаимодействие между разными сценами или узлами?

Вопрос 9.

Как загрузить и инстанцировать сцену динамически во время выполнения игры?

Вопрос 10.

Какие средства профилирования и отладки предоставляет Godot? Как ими пользоваться?

Вопрос 11.

Как реализовать систему сохранения и загрузки данных игры в Godot? Какие существуют подходы и какие классы для этого используются?

Вопрос 12.

Как подключить и использовать Android плагины в Godot? Какие шаги необходимы для интеграции?

Тестовое задание: Разработка мини-игры в Godot

Важно! Выполняется в версии движка 4.5

Задание: Создать простую 2D-игру на движке Godot (**Версия GDScript**), в которой игрок управляет персонажем, собирающим монеты на случайно сгенерированной карте. Цель игры — собрать все монеты, избегая столкновения с врагами, которые перемещаются по карте. Игра может выглядеть так, как вам больше нравится/проще реализовать задание - это может быть top down view или платформер вид сбоку, или другой вариант, все на ваше усмотрение.

Требования:

1. Генерация карты:

- Создать процедурно сгенерированную карту из тайлов. Карта должна состоять из проходимых и непроходимых тайлов (например, стены и земля).
- Размер карты должен быть не менее 20x20 тайлов.
- Для генерации можно использовать шум Перлина или другой алгоритм на усмотрение разработчика.

2. Игровой персонаж:

- Игрок должен управлять персонажем с помощью клавиш (например, стрелок или WASD).
- Персонаж должен плавно перемещаться по карте и сталкиваться с непроходимыми тайлами (стенами).
- Персонаж должен уметь прыгать
 - i. Это может быть Dash
 - ii. Это может быть преодоление препятствия

3. Враги:

- В игре должно быть несколько врагов, которые случайно перемещаются по карте.
- Враги должны уметь избегать непроходимых тайлов и следовать по доступным путям.

- Столкновение с врагом должно завершать игру (например, показом сообщения «Game Over»).
- 4. **Сбор монет:**
 - На карте должно быть случайно размещено несколько монет.
 - При сборе монеты персонажем она исчезает, а счет игрока увеличивается.
 - Когда все монеты собраны, игроку показывается сообщение о победе.
- 5. **Конфигурации:**
 - Проект должен содержать в себе возможность редактировать конфигурации в одном \ нескольких файлах: Скорость передвижения врагов, параметры игрока, условия “Победы” - кол-во собранных монет
- 6. **Интерфейс:**
 - На экране должен отображаться счетчик собранных монет.
 - Добавить кнопки «Начать заново» и «Выйти» после завершения игры (победа или поражение).
- 7. **Сохранение прогресса:**
 - Реализовать возможность сохранения и загрузки прогресса игры, включая положение игрока, врагов и оставшихся монет.
- 8. **Качество кода:**
 - Код должен быть хорошо структурирован, с комментариями, объясняющими ключевые моменты.
 - Применение сигнально-слотовой системы Godot для взаимодействия между объектами будет преимуществом.

Ожидаемый результат:

Завершенная игра с перечисленными выше функциями, предоставленная в виде готового проекта Godot. Проект должен быть легко запускаемым и включать инструкции по управлению.

Оценка:

Оцениваться будет:

- **Функциональность:** Полное соответствие требованиям задания, использование многочисленных возможностей движка.
- **Качество кода:** Чистота, структурированность, комментарии.
- **Оптимизация:** Эффективность кода, работа с производительностью.
- **Креативность:** Подходы к реализации генерации карты, поведение врагов и общий игровой дизайн.